NAVAL POSTGRADUATE SCHOOL
Monterey, California

THESIS

IMPROVING NOMINAL RELIABILITY CONFIDENCE BOUNDS USING COVERAGE PROBABILITIES GENERATED THROUGH MONTE CARLO SIMULATION AND ILLUSTRATED BY MILITARY APPLICATION

by

Captain Jon L. Halverson

September 2000

Thesis Advisor:	David H. Olwell
Second Reader:	Robert R. Read

Approved for public release; distribution is unlimited.

REPORT DOCUMENTATION PAGE

Form Approved
OMB No. 0704-0188

1. AGENCY USE ONLY *(Leave blank)*	2. REPORT DATE	3. REPORT TYPE AND DATES COVERED
	September 2000	Master's Thesis

4. TITLE AND SUBTITLE
Improving Nominal Reliability Confidence Bounds Using Binomial Coverage Probabilities Generated through Monte Carlo Simulation and Illustrated by Military Application

5. FUNDING NUMBERS

6. AUTHOR(S)
Halverson, Jon L.

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)
Naval Postgraduate School
Monterey, CA 93943-5000

8. PERFORMING ORGANIZATION REPORT NUMBER

9. SPONSORING / MONITORING AGENCY NAME(S) AND ADDRESS(ES)
None

10. SPONSORING / MONITORING AGENCY REPORT NUMBER

11. SUPPLEMENTARY NOTES
The views expressed in this thesis are those of the author and do not reflect the official policy or position of the Department of Defense or the U.S. Government.

12a. DISTRIBUTION / AVAILABILITY STATEMENT
Approved for public release; distribution is unlimited.

12b. DISTRIBUTION CODE

13. ABSTRACT

Estimating the failure time of a product with a high degree of confidence is a difficult endeavor. Clearly, if the product is inexpensive and fails quickly, extensive tests can be run to make prediction more accurate. When the item under scrutiny is expensive, not prone to failure, or both, calculating accurate estimates and confidence bounds (CBs) becomes more difficult. Furthermore, many methods currently in use are prone to error, sometimes making a critical part appear more reliable than it actually is. Much of our military uses end-items that fall into this category. The lives of our soldiers, sailors, airmen, and Marines often depend on accurate reliability estimates for the equipment and weapons they work on every day.

This thesis first introduces reliability and the common techniques for measuring it. Secondly, it shows that these estimates are often biased. Next, this bias is quantified using Monte Carlo simulation and corrected through simple tables and equations. The tables and equations can be used to map nominal confidence bounds to actual confidence bounds. Lastly, these results are applied to a Marine Corps program and a test run at a major automotive brake system manufacturer. These examples will illustrate the impact of uncorrected bias and what can be done to correct it.

14. SUBJECT TERMS
Reliability, MLE, Rank Regression, Confidence Bounds, B1, Coverage Probabilities, Monte Carlo Simulation, Weibull.

15. NUMBER OF PAGES
80

16. PRICE CODE

17. SECURITY CLASSIFICATION OF REPORT	18. SECURITY CLASSIFICATION OF THIS PAGE	19. SECURITY CLASSIFICATION OF ABSTRACT	20. LIMITATION OF ABSTRACT
Unclassified	Unclassified	Unclassified	UL

NSN7540-01-280-5500

Standard Form298 (Rev. 2-89)
Prescribed by ANSI Std. 239-18

THIS PAGE INTENTIONALLY LEFT BLANK

Approved for public release; distribution is unlimited.

IMPROVING NOMINAL RELIABILITY CONFIDENCE BOUNDS USING BINOMIAL COVERAGE PROBABILITIES GENERATED THROUGH MONTE CARLO SIMULATION AND ILLUSTRATED BY MILITARY APPLICATION

Jon L. Halverson
Captain, United States Marine Corps
B.S., United States Naval Academy, 1993

Submitted in partial fulfillment of the
requirements for the degree of

MASTER OF SCIENCE IN OPERATIONS RESEARCH

from the

NAVAL POSTGRADUATE SCHOOL
September 2000

Author: Jon L. Halverson

Approved by: David H. Olwell, Thesis Advisor

Robert R. Read, Second Reader

Richard E. Rosenthal, Chairman
Department of Operation Research

THIS PAGE INTENTIONALLY LEFT BLANK

ABSTRACT

Estimating the failure time of a product with a high degree of confidence is a difficult endeavor. Clearly, if the product is inexpensive and fails quickly, extensive tests can be run to make prediction more accurate. When the item under scrutiny is expensive, not prone to failure, or both, calculating accurate estimates and confidence bounds (CBs) becomes more difficult. Much of our military uses end-items that fall into this category. Furthermore, many methods currently in use are prone to error, sometimes making a critical part appear more reliable than it actually is. The lives of our soldiers, sailors, airmen, and Marines often depend on accurate reliability estimates for the equipment and weapons they work on every day.

This thesis first introduces reliability and the common techniques for measuring it. Secondly, it shows that these estimates are often biased. Next, this bias is quantified using Monte Carlo simulation and corrected through simple tables and equations. The tables and equations can be used to map nominal confidence bounds to the actual confidence bounds. Lastly, these results are applied to a Marine Corps program and a test run at a major automotive brake system manufacturer. These examples will illustrate the impact of uncorrected bias and what can be done to correct it.

THIS PAGE INTENTIONALLY LEFT BLANK

THESIS DISCLAIMER

The views expressed in this thesis are those of the author and do not reflect the official policy or position of the Department of Defense or the U.S. Government.

THIS PAGE INTENTIONALLY LEFT BLANK

TABLE OF CONTENTS

LIST OF FIGURES

THIS PAGE INTENTIONALLY LEFT BLANK

LIST OF TABLES

THIS PAGE INTENTIONALLY LEFT BLANK

LIST OF ACRONYMS

AAV	Amphibious Assault Vehicle
AAAV	Advanced Amphibious Assault Vehicle
APG	Aberdeen Proving Grounds
B1	Time When One Percent of the Product Fail
CB	Confidence Bound
CDF	Cumulative Distribution Function
DoD	Department of Defense
MLE	Maximum Likelihood Estimate
MR	Median Rank
MVUE	Minimum Variance Unbiased Estimator
RRX	Rank Regression on X
SEV	Smallest Extreme Value

THIS PAGE INTENTIONALLY LEFT BLANK

ACKNOWLEDGEMENT

I would like to thank Crawford Smith from the ReliaSoft Corporation for the technical support. Your tailored software provided me with crucial feedback and insight.

To Jule Hegwood from Delphi Automotive, I appreciate the problem set and the quick and informative replies to all my questions.

To Sam Buttrey, thank you for getting me started in S-PLUS. I could not have finished this thesis without your help.

To Professor Read, you were always there when I needed help, thank you.

To Professor Whitaker, thank you for guiding me through all the statistics I forgot, and troubleshooting my code. I think it's right now.

To LtCol Dave Olwell, thank you for finding me an interesting problem and guiding me when I struggled with it. You are a role model for students and professors at this school. Best of luck to you and your family in Tucson.

Finally, thanks Mom and Dad for always reminding me that I could work harder on my thesis, and that you wouldn't come to graduation if it wasn't done. I love you.

THIS PAGE INTENTIONALLY LEFT BLANK

EXECUTIVE SUMMARY

Estimating the failure time of a product with a high degree of confidence is a difficult endeavor. Clearly, if the product is inexpensive and fails quickly, extensive tests can be run to make prediction more accurate. When the item under scrutiny is expensive, not prone to failure, or both, calculating accurate estimates and confidence bounds (CBs) becomes more difficult. Much of our military uses end-items that fall into this category. Furthermore, many methods currently in use are prone to error, sometimes making a critical part appear more reliable than it actually is. The lives of our soldiers, sailors, airmen, and Marines often depend on accurate reliability estimates for the equipment and weapons they work on every day.

The Weibull distribution is widely used in reliability estimation because of its versatility. It can describe increasing, constant, and decreasing failure rates. The military and industry alike use the Weibull distribution for estimating reliability and the associated CBs. Rank regression, maximum likelihood estimation, and Bayesian methods are just a few of the tools that exploit the Weibull distribution to achieve these estimates. Past research reveals that while they do provide insight, the nominal CBs use large-sample or asymptotic theory and are often far too optimistic or biased.

> "In practice, this theory is applied to small samples, since crude theory is better than no theory. Confidence bounds are then much too short, but are usually wide enough to be sobering." (Nelson, 1990, pg. 236)

An accurate mapping of the actual CBs to the nominal CBs could significantly reduce this uncertainty. For example, if we want the true 90% lower confidence bound (LCB), we may have to compute a greater LCB using a correction factor. This thesis uses Monte Carlo simulation to explicitly determine coverage for common rank regression and maximum likelihood CB estimation techniques. It shows that the nominal CBs are biased and do not represent the desired confidence. Furthermore, functions are developed to map the actual CB to the desired confidence bound. Lastly, the impact of biased CBs and how these effects should be measured and corrected is illustrated with two examples. The results are applied to a Marine Corps program and a test run at a major automotive brake system manufacturer. These examples will illustrate that the impact of uncorrected bias can result in reliability estimation error as large as 25 percent. The effects of this error are obvious. The military could deploy with systems that don't survive as long as expected and without the necessary logistical support to fix the problem.

I. INTRODUCTION

A. OVERVIEW

The importance of product reliability to the military cannot be overstated. Items needing frequent repair or replacement become increasingly burdensome or unavailable as the military deploys more often and farther from significant logistical support. The issue of reliability has evolved and developed into a key element in competition for military contracts. Since the 1960's, the growth in attention to the reliability of a product has been extensive. Its impact on both the liability of the manufacturer and its efficient and safe use by the operator was recognized. Early in the twentieth century, efforts to study the "survival" of medical patients undergoing different treatments began. In the 1960's, the same science was applied to military and space programs due to demands for more reliable equipment (Nelson, 1992, pg. 3). Now methods are being widely developed for engineering applications to many consumer and industrial products.

Quantifying and accurately estimating reliability is a science, and few people in the military are trained to do it. In this chapter, I first introduce the reader to reliability and some common techniques for measuring it. Second, I show that these techniques are biased, that is, they routinely do not return accurate or true estimates. Lastly, the bias is quantified using Monte Carlo simulation, and corrected using simple tables and equations. It would be advantageous if personnel tasked with acquiring equipment within our military were aware of how reliability is commonly estimated, the shortfalls of the estimates, and finally, how they can be corrected.

B. BACKROUND

The reliability of an item is the *probability* that the item will perform a specified *function*, under specified operational and environmental *conditions*, at and throughout a specified *time*. The word probability represents the random nature of the events that cause a product to fail. Increased reliability is associated with a reduction in the frequency of these random events at a given time (Kales, pg. 7). Before reliability can be tested and measured, the manufacturer and user must agree on what function the product should perform and the conditions under which it will operate. Furthermore, realistic tests must be designed to capture these functions and conditions. Lastly, there must be agreement on how long a product should last. This thesis assumes that the parties involved agreed on and achieved accurate tests, and will focus on the probability and confidence that a product will last for a specified time.

The time at which 1% of the product will fail, on average, is called the B1 time. This is a common measure throughout industry. For a given Weibull distribution, with known parameters η and β, the B1 time is easily calculated from the Weibull cdf:

$$F(t) = .01 = 1 - exp[-(t/\eta)^{\beta}], \qquad (1)$$

then solving for *t*

$$t = \text{B1 time} = \eta * \exp[(1/\beta)\ln(-\ln(1-.01))] = \eta[-\ln(.99)]^{1/\beta}. \qquad (2)$$

When sampling from a Weibull distribution where η and β must be estimated, confidence bounds for B1 are commonly calculated using the t and normal distributions. The quantity t in equation 2, when computed using estimates for η and β, is near the midpoint of its distribution. If the estimates for η and β were unbiased and normally

distributed, the actual t would be less than the B1 time 50% of the time and more than the B1 time 50% of the time. In other words, we are 50% confident that 99% of the product will last until t. In reliability estimation, the user often wants lower bound confidence measurements. The user wants to know how long 99% of the product will last with a certain degree of confidence. For example, the user may ask for the B1 time at 90% confidence. This time is called the B1LCB90. The B1LCB90 will be lower than the B1 time, representing the uncertainty in the estimated B1 time. As the number of samples increases, the proportion of sample based LCBs that will cover the actual value should approach 90% (Figure1, right). For many methods, and small sample sizes, the proportion doesn't approach 90% (Figure 1, left). This thesis attempts to address and remedy this shortfall. All further reference to B1 times and their lower confidence bounds (LCBs) will be represented by B1LCBX, where X is $[(1-\alpha)100\%]$.

When n is small the parameter estimates of η and β are biased, and therefore the B1 estimate is biased. The normal assumption and associated CB calculations become less accurate. The t and normal distributions are still used for estimates because the small sample distributions for η and β have not been derived (Nelson, 1992, pg. 227). Recognizing and defining how the nominal normal based CB coverage differs from the actual CB coverage is central to this thesis.

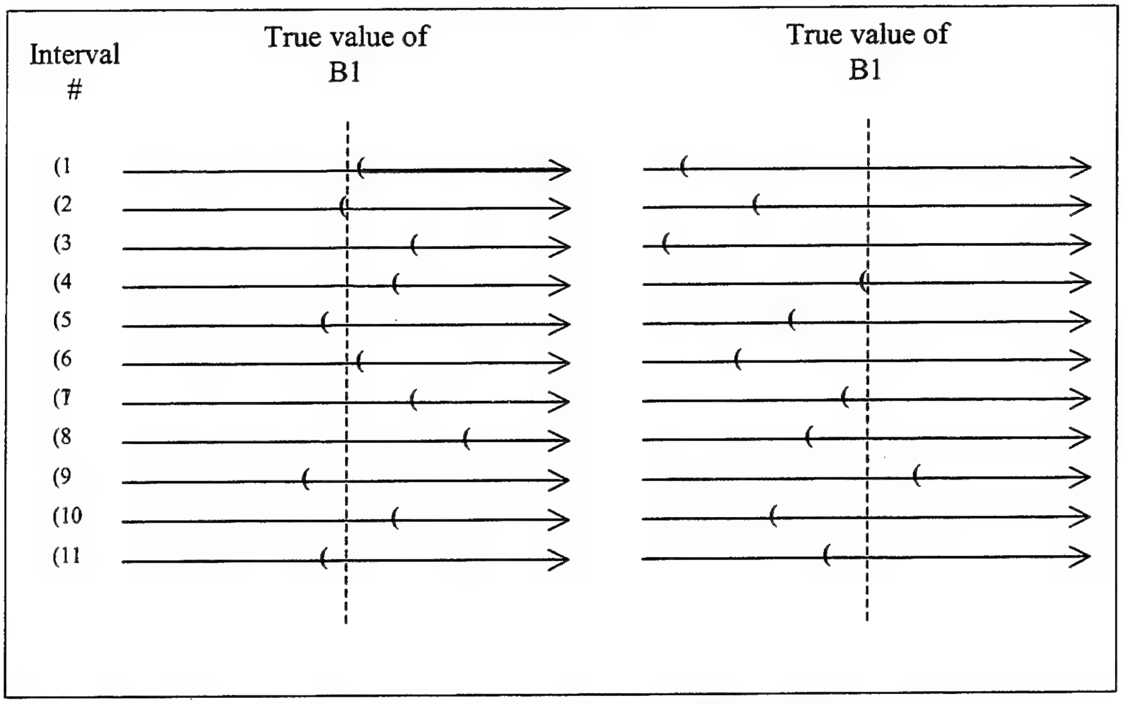

Figure 1. Comparing Biased LCBs with Unbiased LCBs

The proportion of LCB's that cover the B1 time will not approach 90% on the left. Correcting the bias through Monte Carlo simulation will result, ideally, in a proportion approaching 90%.

II. STATISTICAL BACKROUND

A. THE WEIBULL DISTRIBUTION

The two-parameter Weibull distribution is widely used in product reliability testing because it is flexible and therefore fits a variety of data. Its two parameters are the shape parameter (β) and the scale parameter (η). The shape parameter determines the shape of the distribution and represents the rate at which a product fails. The scale parameter determines the characteristic life, specifically, the time at which 63.2 percent of the product fail. The shape parameter is unitless and the scale parameter is in units of time. For the special case where $\beta = 1$, the Weibull is the exponential distribution. For $\beta = 2$, it is a Raleigh distribution. For $3 \leq \beta \leq 4$, the Weibull is very close to the normal and when $\beta \geq 10$ it is close to the smallest extreme value distribution. The range of β and η is the positive real numbers (ReliaSoft, pg. 98).

The Weibull cumulative distribution F(t) is

$$F(t) = 1 - exp[-(t/\eta)^{\beta}] \quad , t \geq 0; \eta, \beta \geq 0$$

where F(t) is the fraction that fail by time t. The reliability function R(t) is

$$R(t) = 1 - F(t) = exp[-(t/\eta)^{\beta}] \quad , t \geq 0; \eta, \beta \geq 0 \qquad (3)$$

where R(t) is the fraction surviving at time t.

The Weibull is related to the smallest extreme value distribution (SEV). This thesis will show that this relationship allows the use of an "ancillary statistic," (Lawless, pg. 147) that is, a statistic whose distribution is independent of the two Weibull parameters, even if the parameters are estimated from the data.

B. THE SMALLEST EXTREME VALUE DISTRIBUTION

Analysis of Weibull data requires familiarity with the smallest extreme value (SEV) distribution. While SEV is used for many types of data, including reliability data, it is mainly of interest because it is related to the Weibull distribution. Weibull data are often analyzed in terms of the simpler SEV because it is a location/scale distribution that allows standardization (Nelson, 1992, pg. 39). Using a simple transformation of the Weibull parameters, a corresponding SEV cumulative distribution function (cdf) is obtained. The SEV cdf

$$F(x) = 1 - exp\{- exp[(x-a)/b]\}; \quad -\infty < x < \infty, \qquad (4)$$

has two parameters, a location parameter *a*, and a scale parameter *b*. Both are transformations of the Weibull parameters ($a = \ln(\eta)$, $b = 1/\beta$). Converting the Weibull cdf into a SEV cdf is rather simple. The steps are

(1) $F(t) = 1 - \exp[-(t/\eta)^{\beta}]$ *transform* $x = ln(t) \Rightarrow t = e^{x}$

(2) $= 1 - \exp[-(e^{x}/\eta)^{\beta}]$

(3) $= 1 - \exp[-(e^{x}/e^{\ln(\eta)})^{\beta}]$ $a = \ln(\eta)$

(4) $= 1 - \exp[-(e^{x}/e^{a})^{\beta}]$

(5) $= 1 - \exp[-(e^{x-a})^{\beta}]$

(6) $= 1 - \exp\{-\exp[(x-a)\beta]\}$ $b = 1/\beta$

(7) $= 1 - \exp\{-\exp[(x-a)/b]\}$ *the SEV cdf*

The standard cdf is

$$\Psi(z) = 1 - exp\{- exp[z]\}; \quad -\infty < x < \infty, \qquad (5)$$

where $z = (x - a)/b$ is called the "standard deviate." $\Psi(z)$ is tabulated by Meeker and Nelson (1974). This conversion is crucial to Weibull data analysis because we can express Weibull data in the following form of

$$F(x) = \Psi[(x-a)/b], \quad -\infty < x < \infty. \quad (6)$$

$$R(x) = 1 - \Psi[(x-a)/b], \quad -\infty < x < \infty. \quad (7)$$

The estimated reliability no longer depends on what the values of *a* and *b* actually are, but on the sample size and censoring mechanism. The use of this standardization is similar to the *t* distribution. Recall that for large *n*, the random variable

$$Z = (x - \mu) / (S/\sqrt{n})$$

has approximately a standard normal distribution. When *n* is small, S is no longer likely to be as close to σ, so the variability increases. Therefore, the distribution of $(x - \mu) / (S/\sqrt{n})$ will be more spread out than the normal. The new distribution is the *t* distribution:

$$T = (x - \mu) / (S/\sqrt{n}).$$

The normal distribution is governed by the mean and standard deviation. A *t* distribution is governed only by the sample size, which determines the degrees of freedom.

Transforming the Weibull into the SEV gives us similar results. The Weibull has two parameters η and β. The SEV reliability distribution $R(z) = 1 - \psi(z)$, where $z = (x-a)/b$, is governed by the sample size, number of failures and the censoring mechanism. R(z) no longer depends on the values of *a* and *b*. Monte Carlo simulations support this result. For given values of sample size *n*, and number of failures *k*, changing η and β doesn't change the estimates and CB's for R(t). For example, two Monte Carlo

simulations of 4000 trials were run for n=3 and k=3. The first simulation used η = 1000 and β = 2 and the second used η = 2 and β = 1000. Both runs calculated B1LCB90. The binomial coverage probability was then calculated. The results are tabled below.

		B1LCB90
β = 2	η = 1000	0.828
β = 1000	η = 2	0.827

Table 1. Comparison of Coverage Probabilities for Different Parameters of the Weibull Distribution

The results illustrate that the B1LCB90 coverage probabilities are independent of the estimated Weibull parameters and therefore support the "ancillary statistic" phenomenon discussed earlier. The results presented in this thesis are applicable to all tests that sample from any assumed 2-parameter Weibull distribution.

C. CENSORING AND SAMPLE SIZE

Constraints on the study of a product's reliability often restrict the observation of exact failure times. Censoring data is a necessity when time or cost limit the length of a reliability study. Time censoring, or "Type I censoring," results when the study ends at a predetermined time before all units have failed. Failure censoring, or "Type II censoring," occurs when the test terminates after a specified number of failures. All items under test fall into one of two categories. The item either failed or was suspended. Suspended means that the item was still working when the test ended, or was removed from the test for reasons other than failure. How items are suspended impacts how the B1LCBX is calculated.

This thesis focuses its research primarily on singly censored and complete data. Data is singly censored when all suspensions occur at the same time, usually after the last failure. Complete data occurs when all items have failed. The techniques are assumed to work for both Type I and Type II censoring. The simulation and estimation techniques can be expanded to multiply censored data. Data is multiply censored when suspensions occur at different points in time. An application with multiply censored data is presented later in this thesis.

Sample size *n* will be defined as the number of units tested. The number of failures *k* must be less than or equal to sample size. When *k* is large, maximum likelihood estimators approach the minimum variance unbiased estimators (MVUE) (Lawless, pg. 291). When *n,* and hence *k,* is small this thesis will show that these asymptotic properties do not hold. This will become clear as this thesis progresses. Table 1 outlines exactly which *n* and *k* combinations that will be studied.

Sample Size	Number of Failures
n	***k***
3	3
6	3
6	4
6	5
6	6
9	3
9	5
9	7
9	9
12	3
12	6
12	9
12	12

Table 2. List of (n,k) Pairs to be Studied

A minimum of three failures is required in order to attain a regression line and variance estimate. This thesis will focus on small sample sizes with complete Type I and Type II data, and singly censored Type I and Type II data.

D. LINEAR RANK REGRESSION

Linear regression requires a straight line be fit to a set of data points such that the sum of squares of the deviation is minimized (ReliaSoft, pg. 39). The data points lie in a plot where the x-axis is the *log(failure time)* and the y-axis is *log(-log(1-R(t)))*. For complete data and right censored data where all suspensions occur after the last failure, the median rank (MR) is used for R(t):

$$MR = 1/(1+((N-j+1)/j)*F_{.5;m;n}) \qquad (8)$$

$$m = 2(N-j+1)$$

$$n = 2*j$$

where $F_{.5;m;n}$ denotes the median of the F distribution with m and n degrees of freedom, for the j^{th} failure out of N items (ReliaSoft Corp, pg. 38). For right censored data with suspensions occurring before the last failure, a failure order number (FON) is calculated. The FON is an adjustment to the median rank and is determined by the location of the suspensions (ReliaSoft Corp, pg. 62). For example, a test is run on three items. The first item is suspended (for reasons other than failure) at $t = 5$ and the second and third failed at $t = 10$, 20 respectively. Simply calculating a median rank would disregard suspension of the first item, even though it might have failed first, second, or third in the interval $5 < t < 20$, had it not been suspended. Calculating a FON accounts for all these possibilities. FON calculation is described in detail in ReliaSoft Corporations's, *Life Data Analysis*

Reference. Once ranks are calculated, the points are plotted and a line is fit. This line yields a slope and intercept from which the time at where the B1 time is easily derived.

E. MAXIMUM LIKELIHOOD ESTIMATION

Maximum likelihood estimation (MLE) is a method of parameter estimation that is independent of data ranks. MLE requires that the distribution be specified and gives the values of the parameters for which the observed sample is most likely to have been generated.

If x is a continuous random variable with a PDF

$$f(\mathbf{x};\theta_1,\theta_2,...\theta_k),$$

where $\theta_1,\theta_2,...\theta_k$ are k unknown constant parameters which are to be estimated and $x_1,x_2,...x_N$ are N independent complete data observations, then the likelihood function is given by

$$L(\theta_1, \theta_2,... \theta_i \,|x_1, x_2,...x_N) = L = \Pi f(x_1; \theta_1, \theta_2,... \theta_i)$$

and the logarithmic likelihood function is

$$\Lambda = \ln L = \sum_{i=1}^{N} \ln f(x_i;\theta_1,\theta_2,...\theta_k).$$

The MLE of $\theta_1,\theta_2,...\theta_k$, are obtained by maximizing either L or Λ (ReliaSoft Corp, pg. 48). Maximizing Λ is computationally easier and the MLE of $\theta_1,\theta_2,...\theta_k$ are the simultaneous solutions of k equations such that

$$\frac{\partial(\Lambda)}{\partial\theta_j} = 0, \quad j = 1,2,...k.$$

The MLE method usually works well with small sample sizes and can converge with only one known failure. It tends to overestimate the parameters but may not converge in the vicinity of the transition point ($\beta = 1$) (Sorrel, pg. 20). As MLEs are only asymptotically unbiased and normally distributed, this thesis will show that BXs based on MLEs can be poorly behaved for small sample sizes.

F. COVERAGE PROBABILITIES VERSUS LEVEL OF CONFIDENCE

Confidence intervals are useful ways to quantify uncertainty due to sampling error arising from limited sample sizes. Confidence intervals have a specified level of confidence (100(1-α)%), typically 90% or 95%, expressing one's confidence (not probability) that a specific interval contains the quantity of interest. It is important to recognize that the confidence level pertains to a probability statement about the performance of the confidence interval *procedure* rather than a statement about any particular interval.

"Coverage probability" is the probability that a confidence interval procedure will result in the interval containing the quantity of interest. Oftentimes, and this thesis will show, the specified "level of confidence" [generically 100(1-α)%] is not equal to the coverage probability. In most practical problems involving censored data, there are no exact confidence interval procedures. This thesis checks the confidence interval approximations of several estimation techniques through simulation. Furthermore, it

provides a correction mechanism to minimize the difference between the level of confidence and the coverage probability.

THIS PAGE INTENTIONALLY LEFT BLANK

III. METHODOLOGY

A. BACKROUND

Determining coverage probabilities requires that the parameters be estimated, and the variance of that estimate be calculated. There are an abundance of software tools that do this. This thesis uses WEIBULL++ and S-PLUS for research, comparison, and simulation. Both software packages do rank regression and maximum likelihood estimation. WEIBULL++ operates in a Windows environment and is very user friendly, but it does not accommodate robust Monte Carlo simulation. S-PLUS is more difficult to use, but it provides more flexibility and simulation power. Furthermore, there are slight differences in how the two use the variance of the estimates. For these reasons, the program used in this thesis was written and implemented in S-PLUS, and WEIBULL++ was used for program validation. The program and remarks can be found in Appendix C.

B. SIMULATION

1. Inputs and Outputs

There are six inputs for the program:

a. Sample Size (n)
b. Number of Failures (k)
c. Weibull Shape Parameter (β)
d. Weibull Scale Parameter (η)
e. Confidence Level ($[1-\alpha]100\%$)
f. Number of Iterations

The first four are self-explanatory. The program iterates through a confidence level vector, which holds the following values:

[.8, .85, .875, .9, .92, .94, .95, .96, .97, .98, .985, .99, .993, .995, .997, .999, .9999,.99999]

Finally, the number of iterations is 1000 and is used for all calculations.

There are three outputs. Each output is a coverage probability associated with different estimation techniques. These techniques are discussed and compared later in this chapter.

2. Generating the Data

The first step in the simulation involves generating and organizing the data. For each iteration, n random variables are drawn from a 2-parameter (β,η) Weibull distribution. Each draw represents a failure time. The times are then ordered in a vector from the lowest value to the highest as in Table 3.

Time
23.4
456.4
81.5
1000.2
213.9
800.7

Time
23.4
81.5
213.9
456.4
800.7
1000.2

Table 3. Ordering Vector of Weibull Values from Lowest to Highest

The last n-k positions in the vector are suspended, that is, they are assigned the time in the k^{th} position. For example if n = 6 and k = 3, see Table 4.

Index	Time	State
1	23.4	Failure
2	81.5	Failure
3	213.9	Failure
4	213.9	Suspended
5	213.9	Suspended
6	213.9	Suspended

Table 4. Reassignment of Suspended Values

For complete data n - k = 0, therefore, there are no suspensions. For right censored data, n > k, and there are n – k suspensions. The result is an ordered vector containing all failures and suspensions.

3. Rank Regression

If the vector contains complete data or right censored data where all suspensions occur after the last failure, median ranks (MR) are calculated. If suspensions occur before the last failure, failure order numbers (FON) are calculated. In order for the regression to be linear, the following transformations are necessary:

$$y = \log(-\log(1\text{-MR})) \text{ or } y = \log(-\log(1\text{-MR(FON)}))$$

$$x = \log(\text{failure time})$$

For each failure from 1 to k, there is an associated (x,y) coordinate. S-Plus then runs regression of x on y in order to calculate reliability estimates. To calculate the B1 time, the time where R(t) = .99, the program asks for the value of x where y = log(-log(.99)). This prediction is based on the regression and is called the fit. Additionally, S-PLUS returns the standard error for that fit based on the uncertainty in the estimates of η and β.

4. Maximum Likelihood Estimation

Calculating MLE in S-PLUS is very straight forward. S-PLUS has a large library of functions for use in survival analysis. Recall the ranked vector created during data generation (Table 3). This program uses the S-PLUS functions "Surv" and "survReg", with the vector, to achieve the R(t) = .99 fit and standard error for that fit.

5. Determining Lower Confidence Bounds

The regression and maximum likelihood calculations each provide a fit and standard error for the fit. Using this data, three different B1LCBX values (Table 5) are calculated for each X in the confidence level vector. The first method (RRX-RRX) used the fit and standard error from rank regression. The second (MLE-MLE) used the fit and standard error from the MLE. The third (RRX-MLE) used the fit from rank regression and standard error from MLE.

Method	Type	Estimate of Parameters	Estimate of Variance	Equation
RRX-RRX	B1LCBX	RRX	RRX	fit.RRX - qt(α,k-2)*se.RRX
MLE-MLE	B1LCBX	MLE	MLE	fit.MLE - qnorm(α)*se.MLE
RRX.MLE	B1LCBX	RRX	MLE	fit.RRX - qt(α,k-2)*se.MLE

Table 5. Equations for Estimating B1LCBX

The t distribution is used for RRX confidence bounds and the normal is used for MLE confidence bounds. The fourth method evaluated in this thesis was calculated in Weibull++. This technique estimated the parameters using rank regression, and then used the parameters in the variance/covariance matrix to determine a MLE estimate of the standard error. This method will be named RRX-MLE W++. The S-PLUS RRX-MLE differs from RRX-MLE W++ in that the latter used the MLE estimate of the

parameters to calculate the standard error. The RRX-MLE W++ is only used when the data is complete data. The W++ simulation software did not accommodate the censoring schemes used in this thesis. Therefore, four methods are compared for complete data and three methods for censored data.

6. Determining Coverage Probabilities

The discussion in CH II, Section B reveals that the coverage probability is independent of β and η. The same results are achieved for any value of β and η. The shape and scale parameters used in the Monte Carlo simulation were 2 and 1000 respectively. The true B1 time is therefore easy to calculate because each draw were samples from a known distribution. Using equation (2)

$$B1 = \eta * exp[(1/\beta)log(-log(1-.01))]$$

$$B1 = 1000 * exp[(1/2)log(-log(.99))]$$

$$B1 = 100.25$$

For each iteration, a count is kept for all four B1LCBX. Each time a B1LCBX is below 100.25 the true B1 is covered and the count is incremented by 1. After 1000 iterations each count is divided by 1000. The resulting fraction is the estimated actual coverage. The confidence level ((1 - α)100%) is the nominal coverage. A truly unbiased method for estimating B1LCBX would result in actual coverage equaling nominal coverage.

THIS PAGE INTENTIONALLY LEFT BLANK

IV. RESULTS

A. INTRODUCTION

The results of the Monte Carlo simulation are presented graphically in this chapter. For each (n, k) pair considered, a graph was developed to illustrate the performance of each estimation technique. The coverage probabilities generated in S-PLUS were exported into ARC (Cook, pg. 3), a statistical package developed at the University of Minnesota. ARC easily displays the data and allows us to smooth it based on ordinary least squares. Smoothing allows us to compare each estimation technique. A sample graph is shown in Figure 1. The x-axis is labeled "nominal coverage" and represents the the confidence level X used in each B1LCBX calculation.

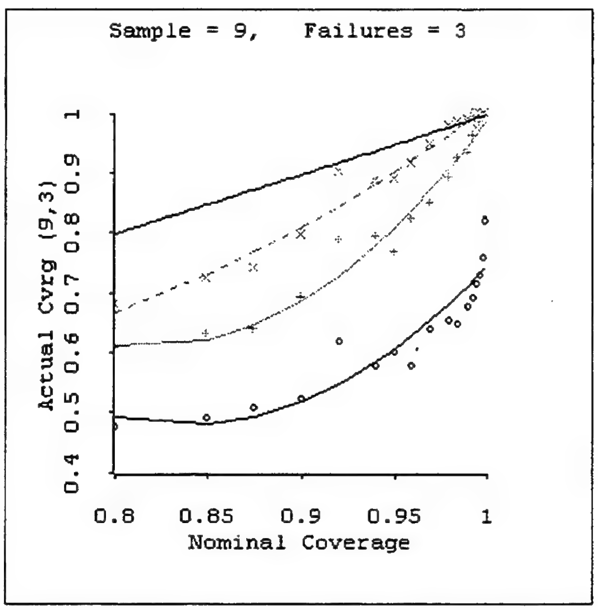

Figure 2. Example Comparison Graph of Nominal Confidence ((1 - α)100%) Versus Actual Coverage Probability for Three Estimation Techniques.

B1LCBX should be below the true B1 time X percent of the time. The y-axis is labeled "Actual Cvrg" and represents the actual coverage determined by the Monte Carlo simulation.

The graph in Figure 1 can be read in two useful ways. First, if the reader wants to know the actual coverage from a given B1LCBX calculation, scan across the x-axis and find the appropriate X. Then trace up the graph until you hit the curve for the method used. Finally, read across to the y-axis. That value is the actual coverage of B1LCBX achieved by the particular estimation technique. For example, using Figure 1 and the estimation technique marked with "+", trace across the x-axis to X = .90. Trace up to the curve and across to the y-axis. That value is approximately .68. Therefore, using this method (+), a B1LCB90 actually covers the true B1 time only 68 percent of the time.

To determine what X is needed to achieve a certain actual coverage, read up the y-axis to the desired actual coverage. Then trace across to the appropriate curve and down to the x-axis. The value on the x-axis is the X required to achieve the desired actual coverage. For example, using Figure 1 and the estimation technique marked with "+", read up the y-axis to the desired actual coverage .90. Then read across to the appropriate curve and down to the x-axis. The value is approximately .975. Therefore, a B1LCB97.5 must be calculated in order to achieve 90 percent actual coverage.

In most cases, the graphs show the reader which estimation technique is least biased, that is, which is closest to the "actual = nominal coverage" line displayed on each graph. Furthermore, the graphs allow for "back-of-the-envelope" adjustments to X so that the B1LCBX calculations give the desired coverage probability. Results are presented for each sample size considered.

B. SAMPLE SIZE AND FAILURES (3,3)

Sample sizes as small as 3 aren't uncommon throughout industry. The expense or vast time required to make certain products fail is significant. For failures equal to 3 the simulation produced consistent results. The combination of rank regression estimation of the parameters and MLE estimation of the variance (RRX-MLE) proved to consistently cover the true B1 time at a rate greater than α. The RRX-MLE method is represented by "x" in Figure 2. Figure 2 shows that although it lies closest to the "y=x" line where nominal coverage equals actual coverage, it covers the actual B1 too often. In fact a B1LCB99 covers B1 100 percent of the time. The same result could be achieved by setting the B1LCB equal to 0. This result provides no useful information and therefore is not be considered. The MLE-MLE and RRX-MLE W++ methods fail to ever achieve 100 percent coverage. A B1LCB100 doesn't cover the true B1 100 percent of the time. We see that the nominal confidence bounds based on such small sample sizes are extremely suspect. The only choice for B1 estimation is RRX-RRX (+). It underestimates coverage but finally reaches 100 percent coverage for sufficiently large X.

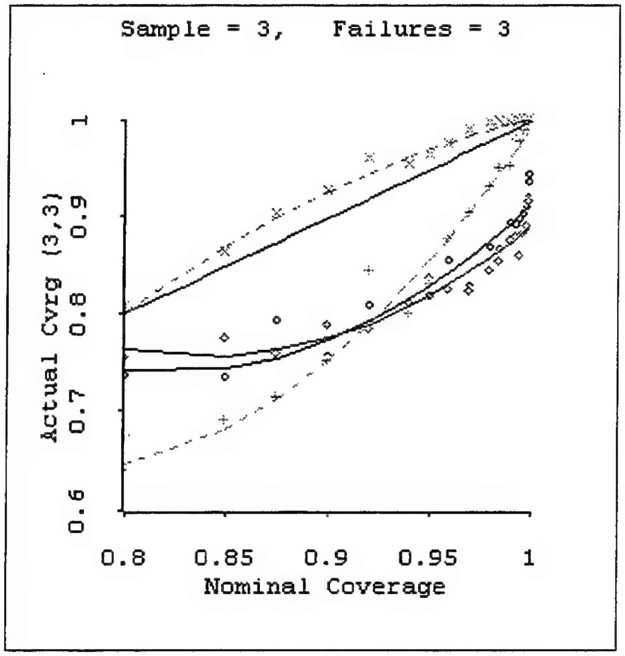

Figure 3. Comparison Graph of Nominal Confidence ([1-α]100%) Versus Actual Coverage Probability for Four Estimation Techniques (n=3, k=3).

RRX-MLE marked by "x", MLE-MLE marked by "o", RXX-RXX marked by "+", RRX-MLE W++ marked by "◊". Based on Monte Carlo samples of size 1000. The x-axis denotes the confidence level (X) used, y-axis denotes actual coverage. The solid line y = x denoting actual coverage equals nominal coverage is included for comparison.

C. SAMPLE SIZE AND FAILURES (6,(3,4,5,6))

For failures equal to 3,4,5, and 6, the simulation produced consistent results. The combination of rank regression estimation of the parameters and MLE estimation of the variance (RRX-MLE) proved to best estimate the B1 time. The RRX-MLE method is represented by "x" in Figure 3. From Figure 3 we can see that in each case, it lies closest to the "y=x" line where actual coverage equals nominal coverage.

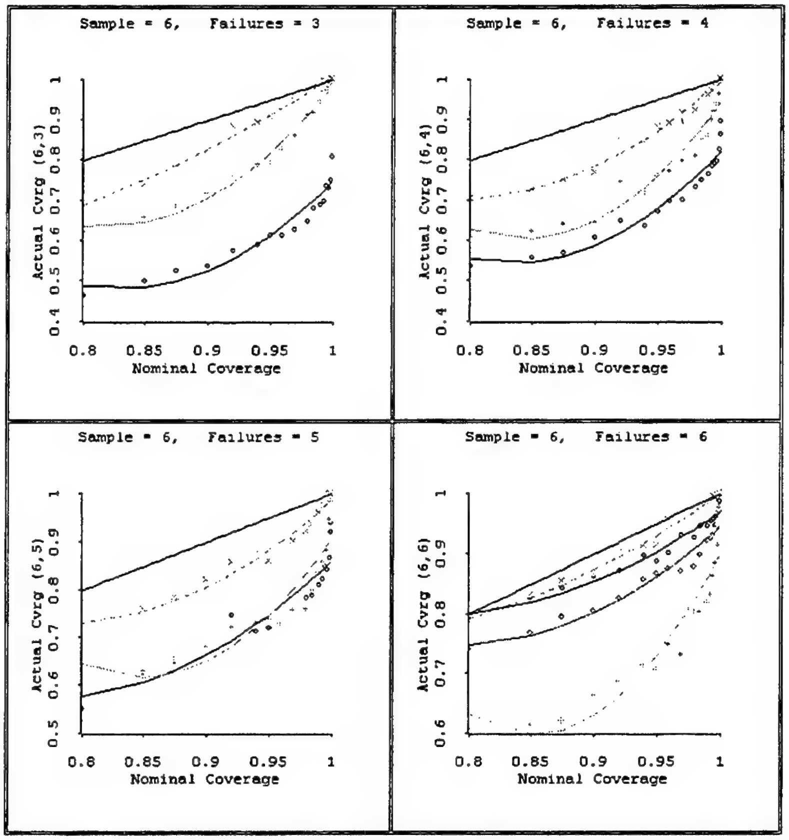

Figure 4. Comparison Graph of Nominal Confidence ([1-α]100%) Versus Actual Coverage Probability for Four Estimation Techniques (n=6, k=3,4,5,6).

RRX-MLE marked by "x", MLE-MLE marked by "o", RXX-RXX marked by "+", RRX-MLE W++ marked by "◊". Based on Monte Carlo samples of size 1000. The x-axis denotes the confidence level (X) used, y-axis denotes actual coverage. The solid line y = x denoting actual equals nominal coverage is included for comparison. Each method fails to achieve the nominal coverage making the product seem more reliable that it actually is. The RRX-MLE method is least biased. Note that the y-axis scale changes from graph to graph.

The RRX-MLE W++ implemented in Weibull++ for complete data (n=6, k=6), performs worse than RRX-MLE and MLE-MLE. Notice that the RRX-RRX method works better for heavily censored data and gets worse as the number of failures *k* approaches the sample size *n* (Figure 4). MLE-MLE performs differently. It estimates poorly for heavily censored data, and better as *k* approaches n (Figure 5).

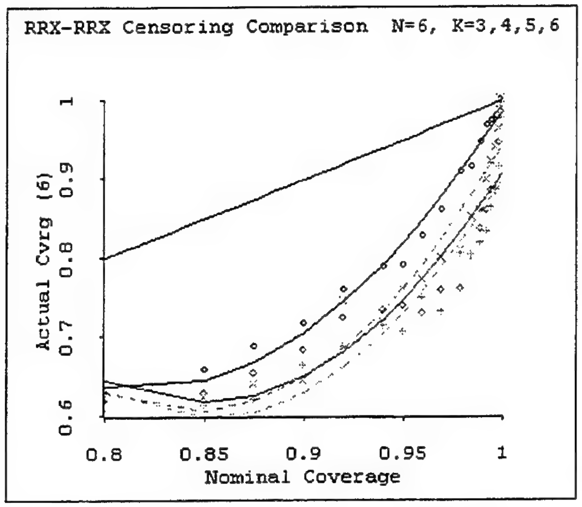

Figure 5. RRX-RRX Comparison of Censoring for Sample of Size 6.

Number of failures marked by 3(o), 4(x), 5(◊), 6(+). As the censoring increases, the coverage probability increases. For example, if the desired coverage is .95, the actual coverage is .73 (k=6), .74 (k=5), .76 (k=4), .81 (k=3).

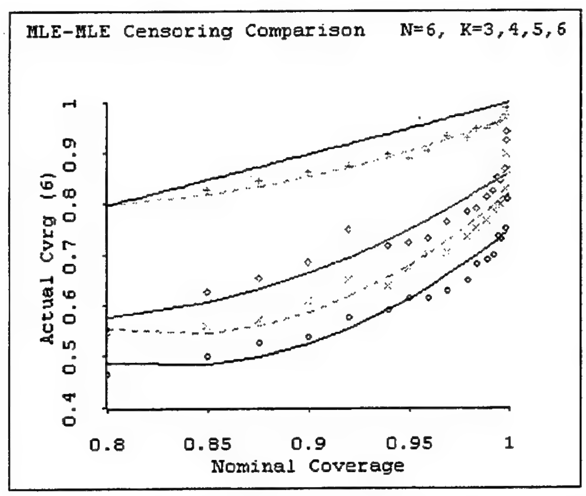

Figure 6. MLE-MLE Comparison of Censoring for Sample of Size 6.

Number of failures marked by 3(o), 4(x), 5(◊), 6(+). As the censoring increases, the coverage probability decreases. For example, if the desired coverage is .95, the actual coverage is approximately .86 (k=6), .73 (k=5), .65 (k=4), .58 (k=3).

D. SAMPLE SIZE AND FAILURES (9, (3,5,7,9))

For failures equal to 3, 5, 7, and 9, the simulation produced consistent results. The combination of rank regression estimation of the parameters and MLE estimation of the variance (RRX-MLE) proved to best estimate the B1 time. The RRX-MLE method is represented by "x" in Figure 6. From Figure 6 we can see that in each case, it lies closest to the "y=x" line where actual coverage equals nominal coverage. The RRX-MLE W++ implemented in Weibull++ for complete data (n=6, k=6), performs worse than RRX-MLE and MLE-MLE. Notice again that the RRX-RRX method works better for heavily censored data and gets worse as the number of failures k approaches the sample size n (Figure 7). MLE-MLE performs differently. It estimates poorly for heavily censored data, and better as k approaches n (Figure 8). Regardless of the value of k, RRX-MLE is the best estimator for B1.

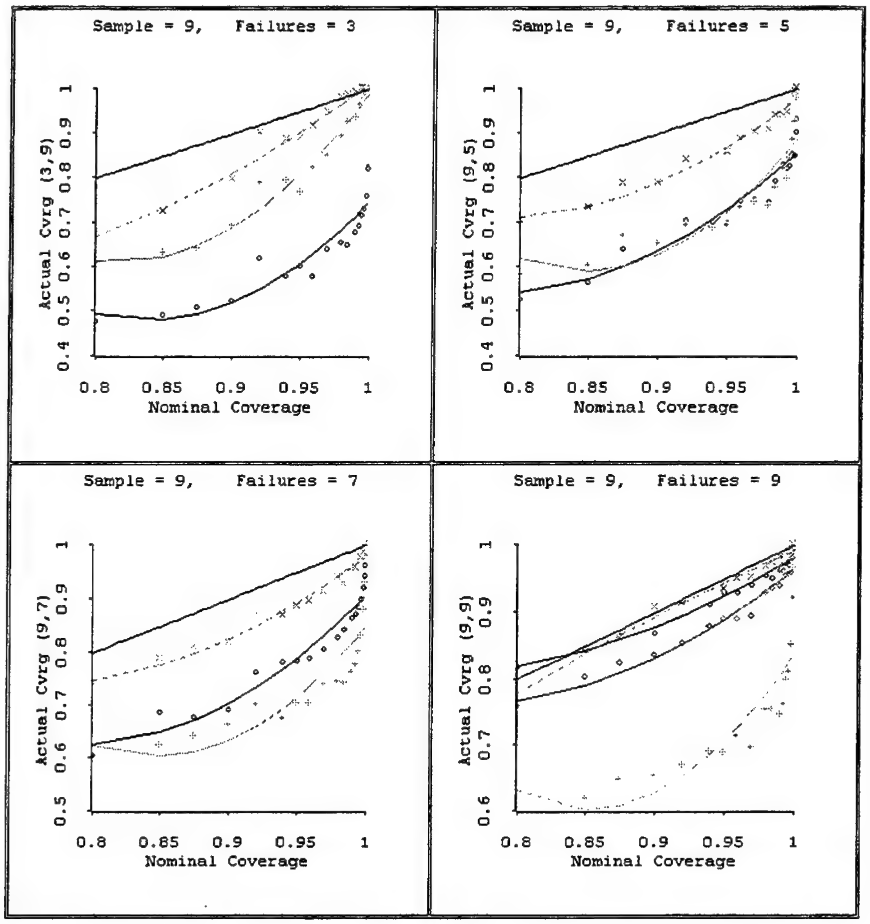

Figure 7. Comparison Graph of Nominal Confidence ([1-α]100%) Versus Actual Coverage Probability for Four Estimation Techniques (n=9, k=3,5,7,9).

RRX-MLE marked by "x", MLE-MLE marked by "o", RXX-RXX marked by "+", RRX-MLE W++ marked by "◊". Based on Monte Carlo samples of size 1000. The x-axis denotes the confidence level (X) used, y-axis denotes actual coverage. The solid line y = x denoting actual equals nominal coverage is included for comparison. Each method fails to achieve the nominal coverage making the product seem more reliable that it actually is. The RRX-MLE method is least biased.

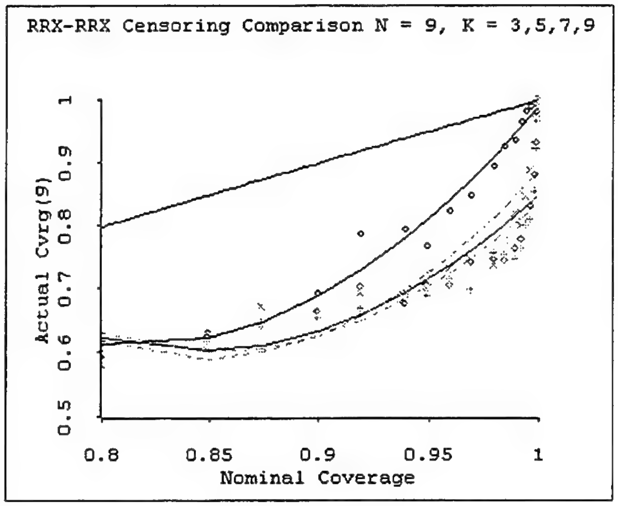

Figure 8. RRX-RRX Comparison of Censoring for Sample of Size 9.

Number of failures marked by 3(o), 5(x), 7(◊), 9(+). As the censoring increases, the coverage probability increases. For example, if the desired coverage is .95, the actual coverage is .71 (k=9), .72 (k=7), .73 (k=5), .82 (k=3).

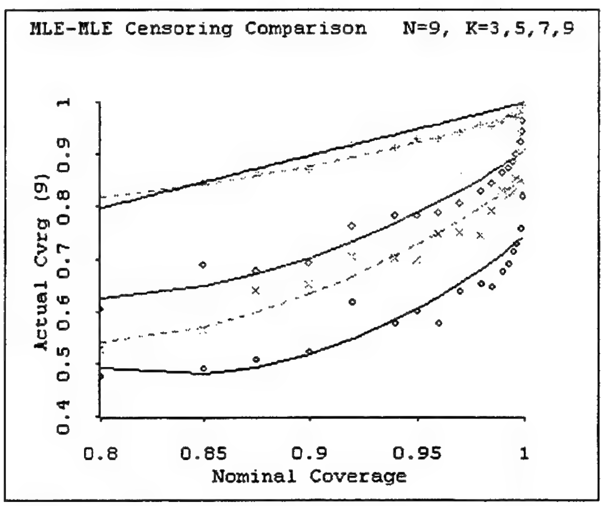

Figure 9. MLE-MLE Comparison of Censoring for Sample of Size 9.

Number of failures marked by 3(o), 5(x), 7(◊), 9(+). As the censoring increases, the coverage probability decreases. For example, if the desired coverage is .95, the actual coverage is .91 (k=9), .80 (k=7), .74 (k=5), .60 (k=3).

E. SAMPLE SIZE AND FAILURES (12,(3,6,9,12))

For failures equal to 3, 6, 9, and 12, the simulation produced consistent results. The combination of rank regression estimation of the parameters and MLE estimation of the variance (RRX-MLE) proved to best estimate the B1 time. The RRX-MLE method is represented by "x" in Figure 9. From Figure 9 we can see that in each case, it lies closest to the "y=x" line where actual coverage equals nominal coverage. The RRX-MLE W++ method implemented in Weibull++ for complete data (n=12, k=12), performs worse than RRX-MLE and MLE-MLE. Notice again that the RRX-RRX method works better for heavily censored data and gets worse as the number of failures *k* approaches the sample size *n* (Figure 10). MLE-MLE performs differently. It estimates poorly for heavily censored data, and better as *k* approaches n (Figure 11). For each value of *k* considered, RRX-MLE proved to be the best estimator for B1. The performance of MLE-MLE approaches that of RRX-MLE for (n =12, k=12). Therefore it is clear that *k* must be greater than 12 for MLE-MLE to be the best estimator for B1.

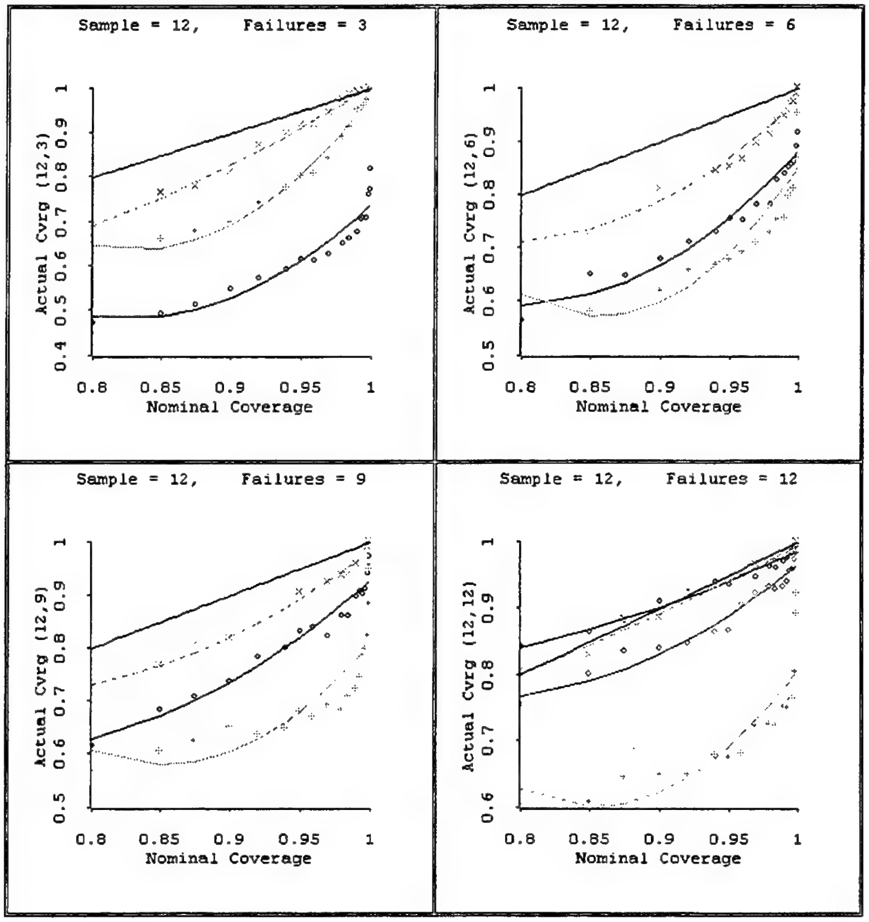

Figure 10. Comparison Graph of Nominal Confidence ([1-α]100%) Versus Actual Coverage Probability for Four Estimation Techniques (n=12, k=3,6,9,12).

RRX-MLE marked by "x", MLE-MLE marked by "o", RXX-RXX marked by "+", RRX-MLE W++ marked by "◊". Based on Monte Carlo samples of size 1000. The x-axis denotes the confidence level (X) used, y-axis denotes actual coverage. The line y = x is included for comparison. Each method fails to achieve the nominal coverage making the product seem more reliable that it actually is. The RRX-MLE method is least biased.

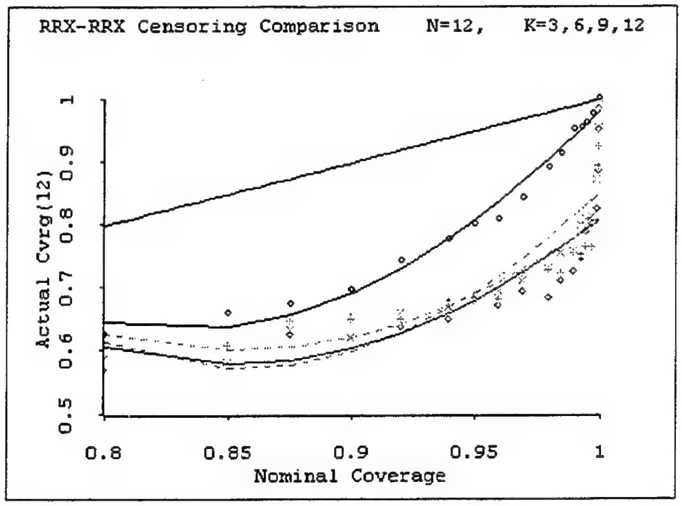

Figure 11. RRX-RRX Comparison of Censoring for Sample of Size 12.

Number of failures marked by 3(o), 6(x), 9(◊), 12(+). As the censoring increases, the coverage probability increases. For example, if the desired coverage is .95, the actual coverage is .66 (k=12), .66 (k=9), .66 (k=6), .81 (k=3).

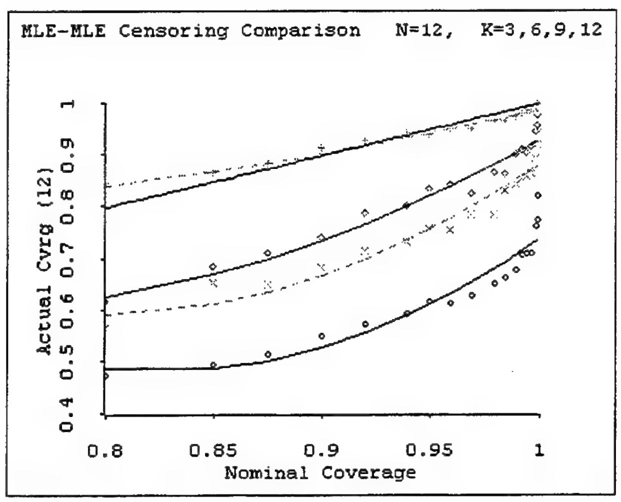

Figure 12. MLE-MLE Comparison of Censoring for Sample of Size 12.

Number of failures marked by 3(o), 6(x), 9(◊), 12(+). As the censoring increases, the coverage probability decreases. For example, if the desired coverage is .95, the actual coverage is .94 (k=12), .80 (k=9), .75 (k=6), .60 (k=3).

V. PRACTICAL APLICATIONS

A. BRAKE SYSTEM APPLICATION

A large automotive parts manufacturer tested a part for a brake system. The test simulated the stresses induced on the brake system by the customer. This company considers the life of the part to be 750,000 cycles. Seven parts were tested, the results are found in Table 6. Six of the seven parts failed. The S-Plus Monte Carlo simulation was

Sample	# Completed Cycles	Failed (F) or Suspended (S)
1	822000	F
2	831675	F
3	849633	F
4	887862	F
5	887862	F
6	901713	F
7	901713	S

Table 6. Data from the Results of the Brake Test

run for n = 7 and k = 6. A comparison of the results for each estimation technique is found in Figure 12. Clearly, the RRX-MLE methods is the least biased estimator for B1. A second order polynomial was then fit to the results of the Monte Carlo simulation (Table 7) for the RRX-MLE method.

$$Y = -2.0547*X^2 + 4.3094*X - 1.2528$$

This equation can now be used to map the nominal confidence to the actual confidence. The manufacturer wanted the number of cycles where 99% reliability at 90% confidence (B1LCB90) is attained. Table 7 shows that when using RRX-MLE, B1LCB90 only

covers the true value 81.2% of the time. The equation below will tell us what level of confidence is necessary to achieve 90% coverage probability.

$$Y = -2.0547*(.90)^2 + 4.3094*(.90) - 1.2528$$

$$Y = .961$$

In order to achieve a nominal B1LCB90 we must actually do a B1LCB96.1 calculation.

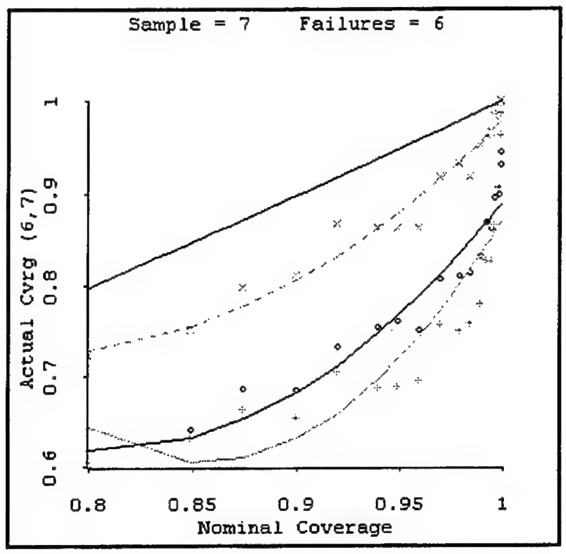

Figure 13. Comparison Graph of Nominal Confidence ([1-α]100%) Versus Actual Coverage Probability for Four Estimation Techniques (n=7, k=6).

RRX-MLE marked by "x", MLE-MLE marked by "o", RXX-RXX marked by "+", RRX-MLE W++ marked by "◊". Based on Monte Carlo samples of size 1000. The x-axis denotes the confidence level (X) used, y-axis denotes actual coverage. The line y = x is included for comparison.

For this example, using RRX-MLE, the fit for B1 is 746275 cycles. The RRX-MLE estimate of B1LCB90 is 685515 cycles. According to the Monte Carlo simulation, the B1LCB90 will be below the true B1 only 81.2% of the time. The estimate of B1LCB96.1 is 654976. According to Monte Carlo simulation, the B1 time will occur after 654976 cycles with a frequency approaching 90%.

Nominal Coverage	Monte Carlo Sim Actual Coverage
0.8	0.723
0.85	0.751
0.875	0.799
0.9	0.812
0.92	0.868
0.94	0.864
0.95	0.863
0.96	0.863
0.97	0.919
0.98	0.932
0.985	0.919
0.99	0.953
0.993	0.962
0.995	0.965
0.997	0.983
0.999	0.991
0.9999	0.998

Table 7. Comparison of Actual Coverage Probability to Nominal Confidence ([1-α]100%) for 6 Failures out of 7 Samples.

Actual coverage determined by smoothing the data using second order polynomials fit to Monte Carlo simulation results.

Using 2-parameter Weibull analysis, the results show that the expected life of 750,000 cycles is unrealistic. The company elected to conduct 3-parameter Weibull analysis because this distribution fit their data more appropriately. This example also shows that the S-Plus code will expand and apply to any (n,k) pair.

B. ADVANCED AMPHIBIOUS ASSAULT VEHICLE APPLICATION

The Advanced Amphibious Assault Vehicle (AAAV) is the Marine Corps' upgrade to the Amphibious Assault Vehicle (AAV). Like the AAV, the AAAV is an armored vehicle used to move Marines safely over both ground and water. Instead of wheels, it moves on two tracks much like a tank. The tracks are kept on the system that turns them by objects called "centerguides." There are 102 centerguides per track.

Analysts performed Weibull analysis of centerguide failure/survival results. The data was obtained from recent testing performed at the Aberdeen Test Center, Aberdeen Proving Grounds (APG), Maryland. The data is displayed in Table 8. The analysts estimate the parameters, η and β, of a two-parameter Weibull distribution using MLE. They estimate $\eta = 4600$ and $\beta = 5.1$ (US Army Material Systems Analysis Activity, pg. 19). Using equation 2, the point estimate for B1 is

$$B1 = 4600 * exp[(1/5.1) * log(-log(1-.01))] = 2895.9 \text{ miles.}$$

This estimate of B1 is the same achieved with the program developed in this thesis. Next, the analysts calculate a LCB80 on each parameter. The estimates are $\eta = 4077.01$ and $\beta = 3.9527$. From equation 2 the B1LCB80 is

$$B1 = 4077 * exp[(1/3.9527) * log(-log(1-.01))] = 2244.1 \text{ miles.}$$

The S-Plus program uses the MLE estimate of both the fit and variance to validate that the estimation methods used at APG are the same as the methods used here. The S-Plus code returns the same answer, B1LCB80 = 2244.1 miles. The simulation is then run using MLE-MLE estimation to compare nominal confidence levels to the actual coverage probability. The simulation sets $n = 204$ and $k = 23$ where each failure occurs in the same position as the test set (Table 8) for each iteration. The results are shown in Table 9.

Miles Run	Failed (F) or Suspended (S)	Miles Run	Failed (F) or Suspended (S)	Miles Run	Failed (F) or Suspended (S)	Miles Run	Failed (F) or Suspended (S)
2021	F	3049	S	3049	S	3049	S
2021	F	3049	S	3049	S	3049	S
2021	F	3049	S	3049	S	3049	S
2021	F	3049	S	3049	S	3049	S
2021	F	3049	S	3049	S	3049	S
2021	F	3049	S	3049	S	3049	S
2021	F	3049	S	3049	S	3049	S
2337	F	3049	S	3049	S	3049	S
2337	S	3049	S	3049	S	3049	S
2337	S	3049	S	3049	S	3049	S
2523	F	3049	S	3049	S	3049	S
2523	F	3049	S	3049	S	3049	S
2523	F	3049	S	3049	S	3049	S
2523	F	3049	S	3049	S	3049	S
2523	F	3049	S	3049	S	3049	S
2523	F	3049	S	3049	S	3049	S
2523	F	3049	S	3049	S	3049	S
2833	F	3049	S	3049	S	3049	S
2833	F	3049	S	3049	S	3049	S
3049	F	3049	S	3049	S	3049	S
3049	F	3049	S	3049	S	3049	S
3049	F	3049	S	3049	S	3049	S
3049	F	3049	S	3049	S	3049	S
3049	F	3049	S	3049	S	3049	S
3049	F	3049	S	3049	S	3049	S
3049	S	3049	S	3049	S	3049	S
3049	S	3049	S	3049	S	3049	S
3049	S	3049	S	3049	S	3049	S
3049	S	3049	S	3049	S	3049	S
3049	S	3049	S	3049	S	3049	S
3049	S	3049	S	3049	S	3049	S
3049	S	3049	S	3049	S	3049	S
3049	S	3049	S	3049	S	3049	S
3049	S	3049	S	3049	S	3049	S
3049	S	3049	S	3049	S	3049	S
3049	S	3049	S	3049	S	3049	S
3049	S	3049	S	3049	S	3049	S
3049	S	3049	S	3049	S	3049	S
3049	S	3049	S	3049	S	3049	S
3049	S	3049	S	3049	S	3049	S
3049	S	3049	S	3049	S	3049	S
3049	S	3049	S	3049	S	3049	S
3049	S	3049	S	3049	S	3049	S
3049	S	3049	S	3049	S	3049	S
3049	S	3049	S	3049	S	3049	S
3049	S	3049	S	3049	S	3049	S
3049	S	3049	S	3049	S	3049	S
3049	S	3049	S	3049	S	3049	S
3049	S	3049	S	3049	S	3049	S
3049	S	3049	S	3049	S	3049	S
3049	S	3049	S	3049	S	3049	S

Table 8. Data from Results of AAAV Centerguide Test.

Table 9 shows us that the B1LCB80 previously calculated will lie below the true

B1 only 58.7% of the time. From Table 9 we can see that in order to get 80% coverage, we will have to compute roughly a B1LCB95. Fitting a second order polynomial to the tabled data gives us a more accurate estimate.

$$Y = -.8104*(.80)^2 + 1.8119*(.80) + .0073 = .938$$

Nominal Coverage	Monte Carlo Sim Actual Coverage
0.8	0.587
0.85	0.685
0.875	0.699
0.9	0.719
0.92	0.792
0.94	0.814
0.95	0.801
0.96	0.84
0.97	0.86
0.98	0.888
0.985	0.897
0.99	0.915
0.993	0.931
0.995	0.93
0.997	0.941
0.999	0.961
0.9999	0.979
0.99999	0.99

Table 9. Comparison of Actual Coverage Probability to Nominal Confidence ([1-α]100%) for 23 Ordered Failures out of 204 Samples.

Actual coverage determined by smoothing the data using a second order polynomial fit to Monte Carlo simulation results.

A B1LCB93.8 is required for 80% actual coverage. B1LCB93.8 is 1597 miles. The results of the Monte Carlo simulation state that the true B1 will occur after 1597 miles, 80% of the time. Comparing 1597 miles with 2244 miles shows us that estimating B1LCB80 with an assumed 2-parameter Weibull distribution using MLE-MLE, the estimate is 25% too optimistic. This can have drastic effects when Marine Corps units

deploy or engage in long-term operations. The results illustrate that the AAAV may not last as long as expected. Without this knowledge, logistical planners may underestimate the number of spares required to keep this system operating at the necessary level.

THIS PAGE INTENTIONALLY LEFT BLANK

VI. CONCLUSIONS AND RECOMMENDATIONS

A. CONCLUSIONS

The results of this thesis are clear. The most common methods for estimating B1 LCBs yield biased results, making the item under test appear more reliable that is actually is. The method for estimating least-biased B1 LCBs is a combination of rank regression and maximum likelihood estimation (RRX-MLE) using the wider t distribution for confidence bound calculations. The only case where this doesn't hold is $n = 3$ and $k = 3$. For this case, rank regression is recommended. No software package, to my knowledge, uses this combination (RRX-MLE) for B1 LCB estimation.

When comparing straight rank regression (RRX-RRX) and straight maximum likelihood estimation (MLE-MLE), rank regression estimates B1 better for heavily censored data, and MLE estimates B1 better as k approaches n. Correction factors, based on Monte Carlo simulation, are the only means for these methods to accurately estimate B1. If one doesn't have correction factors available, this thesis clearly shows which estimation technique is least biased for each (n,k) pair studied.

The results also clearly show that some methods cannot predict B1 with high levels of confidence. The simulation results yield cases where the coverage probability doesn't converge to 1. These estimation techniques should not be used for those cases.

Errors from Monte Carlo simulation can be reduced by increasing the number of trials, or fitting smooth functions to the results. Increasing the number of trials wasn't feasible for this thesis. After selecting the most appropriate estimation technique for each (n, k) pair, curves were fit to second order polynomials using ordinary least squares. The

equations were used to calculate the nominal confidence levels required to achieve common confidence levels used throughout the DoD and industry. The equations and nominal values appear in Table 10. If the recommended methods in Table 10 cannot

				Confidence Level Needed to Achieve X			
n	**k**	**Method**	**Equations for Nominal Confidence**	X = .80	X = .90	X = .95	X = .99
3	3	RRX-RRX	$Y = -1.0203X^2+2.2017X-.1834$	0.9250	0.9717	0.9874	0.9963
6	3	RRX-MLE	$Y = -.4786X^2+1.4325X+.0422$	0.8800	0.9438	0.9711	0.9913
6	4	RRX-MLE	$Y = -1.33X^2+2.865X-.536$	0.9050	0.9652	0.9854	0.9968
6	5	RRX-MLE	$Y = -2.022X^2+4.2056X-1.1831$	0.8873	0.9641	0.9874	0.9900
6	6	RRX-MLE	$Y = -1.9021X^2+4.3544-1.45$	0.8162	0.9283	0.9700	0.9966
9	3	RRX-MLE	$Y = -.4428X^2+1.3044X+.1326$	0.8927	0.9479	0.9722	0.9900
9	5	RRX-MLE	$Y = -1.733X^2+3.6177X-.8831$	0.9019	0.9691	0.9897	0.9999
9	7	RRX-MLE	$Y = -2.0468X^2+4.3546X-1.3038$	0.8699	0.9574	0.9858	0.9999
9	9	RRX-MLE	$Y = .6034X^2-.1355X+.5401$	0.8179	0.9069	0.9559	0.9973
12	3	RRX-MLE	$Y = -.4958X^2+1.4681X+.0238$	0.8810	0.9435	0.9710	0.9913
12	6	RRX-MLE	$Y = -1.5926X^2+3.3687-.7738$	0.9019	0.9680	0.9891	0.9999
12	9	RRX-MLE	$Y = -1.5921X^2+3.4895X-.8943$	0.8784	0.9566	0.9839	0.9999
12	12	RRX-MLE	$Y = -.2974X^2+1.5453X-.2429$	0.8030	0.9070	0.9567	0.9955

Table 10. Equations for Mapping Nominal Confidence to Actual Coverage Using Second Order Polynomials.

For each (n,k) pair, the least biased method is accompanied by an equation that maps nominal or "desired" coverage ([1-α]100%) to actual coverage probability. Based on Monte Carlo samples of size 1000. The desired confidence, or nominal coverage, is represented by X. The required confidence (X) to get the nominal coverage is represented by Y.

be used, Appendix A and B display the data for each method considered. Curves can be fit for those methods, provided they reasonably fit the data and converge to 1.

The thesis clearly shows that when using common techniques for estimating survivability of military systems and equipment, caution must be used.

Miscalculation and error can have drastic effects on the timely logistical support and operational tempo of our deploying units.

B. RECOMMENDATIONS

The following recommendations can enhance the research and results found in this thesis:

- Increase the number of Monte Carlo trials for each (n,k) pair from 1000 to 10,000. This may require another software package.
- Expand rank regression code to fully accommodate multiply censored data. The current code can only use MLE-MLE for multiply censored data. It will not allow for RRX when an item is suspended before the first failure.
- Apply a variance reduction scheme to the program. For each (n, k) pair the current program estimates each B1LCBX from different data. Using one data set for all B1LCBX calculations will smooth and speed up the results.
- Use Monte Carlo simulation for Non-Parametric and Bayesian estimation methods. Compare their coverage probabilities to the results presented here.
- Use Monte Carlo simulation for 3-parameter Weibull distributions. The failure-free parameter fits many types of data very well.
- Publicize the results of this thesis so that DoD will understand the possible effects of inherent bias when using common estimation techniques on small sample sizes and failures.

- Incorporate the applications into OA3101, OA3102, and OA3103 so that students might gain a greater appreciation of the power of Monte Carlo simulation and the practical limits of asymptotic theory.
- Adjust military reliability standards to require actual coverage calculations.

Further work in this subject area is needed. The effect it can have on military readiness is significant. Our forces must operate reliable and safe systems in the conduct of their mission.

APPENDIX A. MONTE CARLO SIMULATIONS RESULTS FOR RRX-RRX AND MLE-MLE

Nominal	Actual Coverage													
Coverage	**Method**	**(3,3)**	**(3,6)**	**(4,6)**	**(5,6)**	**(6,6)**	**(3,9)**	**(5,9)**	**(7,9)**	**(9,9)**	**(3,12)**	**(6,12)**	**(9,12)**	**(12,12)**
0.8	mle-mle	0.736	0.466	0.538	0.553	0.792	0.476	0.527	0.605	0.819	0.473	0.566	0.618	0.843
0.85	mle-mle	0.734	0.5	0.56	0.627	0.828	0.492	0.566	0.689	0.843	0.494	0.651	0.685	0.865
0.875	mle-mle	0.794	0.528	0.569	0.652	0.844	0.509	0.641	0.679	0.866	0.516	0.649	0.711	0.883
0.9	mle-mle	0.757	0.539	0.608	0.683	0.863	0.523	0.652	0.692	0.869	0.551	0.682	0.739	0.912
0.92	mle-mle	0.809	0.575	0.649	0.749	0.873	0.62	0.705	0.763	0.916	0.574	0.713	0.785	0.925
0.94	mle-mle	0.813	0.591	0.636	0.716	0.898	0.578	0.702	0.782	0.911	0.594	0.732	0.801	0.941
0.95	mle-mle	0.819	0.613	0.673	0.723	0.888	0.602	0.698	0.784	0.93	0.616	0.756	0.833	0.936
0.96	mle-mle	0.856	0.613	0.699	0.732	0.902	0.579	0.748	0.79	0.929	0.615	0.754	0.842	0.949
0.97	mle-mle	0.828	0.628	0.702	0.762	0.932	0.639	0.752	0.807	0.94	0.629	0.782	0.824	0.948
0.98	mle-mle	0.868	0.649	0.734	0.784	0.928	0.655	0.744	0.829	0.955	0.652	0.783	0.864	0.964
0.985	mle-mle	0.867	0.682	0.752	0.789	0.946	0.649	0.791	0.843	0.951	0.665	0.83	0.862	0.963
0.99	mle-mle	0.894	0.691	0.765	0.813	0.946	0.677	0.83	0.866	0.962	0.677	0.84	0.9	0.972
0.993	mle-mle	0.893	0.699	0.786	0.824	0.955	0.694	0.821	0.873	0.973	0.708	0.852	0.909	0.978
0.995	mle-mle	0.898	0.735	0.792	0.85	0.958	0.717	0.826	0.884	0.973	0.711	0.857	0.903	0.986
0.997	mle-mle	0.904	0.73	0.797	0.843	0.962	0.732	0.853	0.899	0.97	0.709	0.863	0.913	0.983
0.999	mle-mle	0.912	0.75	0.828	0.867	0.972	0.761	0.849	0.922	0.987	0.763	0.891	0.942	0.99
0.9999	mle-mle	0.936	0.811	0.865	0.922	0.985	0.823	0.902	0.943	0.992	0.775	0.918	0.955	0.997
0.99999	mle-mle	0.944	0.809	0.898	0.94	0.988	0.821	0.931	0.963	0.994	0.821	0.952	0.974	0.995

Nominal	Actual Coverage													
Coverage	**Method**	**(3,3)**	**(3,6)**	**(4,6)**	**(5,6)**	**(6,6)**	**(3,9)**	**(5,9)**	**(7,9)**	**(9,9)**	**(3,12)**	**(6,12)**	**(9,12)**	**(12,12)**
0.8	**rx-rx**	0.64	0.623	0.609	0.619	0.607	0.607	0.582	0.594	0.604	0.628	0.585	0.57	0.604
0.85	**rx-rx**	0.69	0.659	0.623	0.629	0.613	0.632	0.602	0.624	0.618	0.662	0.582	0.606	0.609
0.875	**rx-rx**	0.714	0.688	0.64	0.653	0.62	0.641	0.67	0.643	0.647	0.677	0.632	0.625	0.646
0.9	**rx-rx**	0.751	0.717	0.644	0.682	0.664	0.692	0.653	0.663	0.654	0.699	0.62	0.651	0.65
0.92	**rx-rx**	0.843	0.76	0.745	0.723	0.687	0.788	0.693	0.703	0.668	0.743	0.66	0.636	0.649
0.94	**rx-rx**	0.799	0.79	0.718	0.733	0.713	0.795	0.686	0.676	0.69	0.777	0.668	0.649	0.678
0.95	**rx-rx**	0.831	0.792	0.76	0.739	0.705	0.769	0.692	0.705	0.688	0.802	0.679	0.68	0.675
0.96	**rx-rx**	0.877	0.828	0.772	0.73	0.748	0.825	0.733	0.705	0.714	0.81	0.694	0.672	0.681
0.97	**rx-rx**	0.903	0.861	0.799	0.759	0.731	0.849	0.746	0.742	0.697	0.843	0.711	0.694	0.723
0.98	**rx-rx**	0.931	0.91	0.809	0.76	0.804	0.894	0.737	0.746	0.755	0.891	0.729	0.684	0.726
0.985	**rx-rx**	0.951	0.916	0.851	0.8	0.802	0.927	0.776	0.744	0.754	0.915	0.753	0.709	0.723
0.99	**rx-rx**	0.952	0.946	0.86	0.835	0.818	0.935	0.82	0.763	0.747	0.952	0.756	0.724	0.753
0.993	**rx-rx**	0.98	0.968	0.899	0.859	0.833	0.965	0.799	0.778	0.763	0.955	0.797	0.747	0.75
0.995	**rx-rx**	0.978	0.973	0.921	0.885	0.863	0.982	0.845	0.801	0.799	0.962	0.806	0.787	0.763
0.997	**rx-rx**	0.99	0.98	0.939	0.882	0.885	0.988	0.886	0.831	0.81	0.976	0.811	0.8	0.764
0.999	**rx-rx**	0.995	0.99	0.963	0.945	0.913	0.995	0.927	0.879	0.852	0.994	0.87	0.825	0.804
0.9999	**rx-rx**	0.998	0.998	0.997	0.984	0.975	0.999	0.978	0.93	0.921	1	0.952	0.885	0.893
0.99999	**rx-rx**	1	1	1	0.999	0.995	1	0.996	0.979	0.966	1	0.988	0.95	0.924

THIS PAGE INTENTIONALLY LEFT BLANK

APPENDIX B. MONTE CARLO SIMULATIONS RESULTS FOR RRX-MLE AND RRX-MLE W++

Nominal		Actual Coverage												
Coverage	**Method**	**(3,3)**	**(3,6)**	**(4,6)**	**(5,6)**	**(6,6)**	**(3,9)**	**(5,9)**	**(7,9)**	**(9,9)**	**(3,12)**	**(6,12)**	**(9,12)**	**(12,12)**
0.8	**rx-mle**	0.807	0.701	0.703	0.723	0.79	0.682	0.703	0.737	0.772	0.692	0.708	0.726	0.804
0.85	**rx-mle**	0.863	0.738	0.725	0.761	0.832	0.725	0.733	0.789	0.85	0.765	0.722	0.768	0.83
0.875	**rx-mle**	0.903	0.779	0.751	0.785	0.855	0.743	0.79	0.813	0.862	0.781	0.774	0.806	0.884
0.9	**rx-mle**	0.927	0.825	0.769	0.821	0.867	0.798	0.789	0.82	0.907	0.811	0.813	0.819	0.887
0.92	**rx-mle**	0.961	0.884	0.855	0.857	0.909	0.902	0.84	0.876	0.921	0.874	0.851	0.847	0.921
0.94	**rx-mle**	0.955	0.893	0.845	0.876	0.914	0.889	0.842	0.87	0.934	0.9	0.847	0.868	0.928
0.95	**rx-mle**	0.964	0.909	0.881	0.861	0.913	0.892	0.86	0.886	0.934	0.92	0.854	0.907	0.949
0.96	**rx-mle**	0.976	0.931	0.891	0.882	0.956	0.916	0.887	0.896	0.951	0.919	0.867	0.909	0.949
0.97	**rx-mle**	0.989	0.958	0.916	0.901	0.949	0.949	0.901	0.914	0.952	0.947	0.898	0.927	0.967
0.98	**rx-mle**	0.998	0.976	0.924	0.921	0.969	0.981	0.907	0.94	0.967	0.974	0.915	0.939	0.969
0.985	**rx-mle**	0.999	0.982	0.955	0.936	0.978	0.988	0.94	0.929	0.973	0.986	0.939	0.937	0.973
0.99	**rx-mle**	0.997	0.997	0.962	0.961	0.987	0.993	0.941	0.958	0.982	0.993	0.947	0.959	0.983
0.993	**rx-mle**	1	0.995	0.971	0.969	0.99	0.998	0.945	0.958	0.978	0.998	0.957	0.964	0.981
0.995	**rx-mle**	0.999	0.995	0.982	0.984	0.995	1	0.962	0.966	0.99	0.995	0.971	0.969	0.988
0.997	**rx-mle**	1	1	0.99	0.977	0.993	1	0.985	0.978	0.989	0.999	0.975	0.975	0.988
0.999	**rx-mle**	1	1	0.996	0.998	0.997	1	0.993	0.986	0.996	1	0.988	0.992	0.996
0.9999	**rx-mle**	1	1	1	0.999	1	1	1	0.997	1	1	0.999	0.997	1
0.99999	**rx-mle**	1	1	1	1	1	1	1	1	1	1	1	1	1

Nominal		Actual Coverage												
Coverage	**Method**	**(3,3)**	**(3,6)**	**(4,6)**	**(5,6)**	**(6,6)**	**(3,9)**	**(5,9)**	**(7,9)**	**(9,9)**	**(3,12)**	**(6,12)**	**(9,12)**	**(12,12)**
0.8	**rx-mle W++**	0.755	NA	NA	NA	0.74	NA	NA	NA	0.758	NA	NA	NA	0.755
0.85	**rx-mle W++**	0.774	NA	NA	NA	0.768	NA	NA	NA	0.802	NA	NA	NA	0.8
0.875	**rx-mle W++**	0.758	NA	NA	NA	0.796	NA	NA	NA	0.825	NA	NA	NA	0.835
0.9	**rx-mle W++**	0.788	NA	NA	NA	0.805	NA	NA	NA	0.835	NA	NA	NA	0.84
0.92	**rx-mle W++**	0.784	NA	NA	NA	0.826	NA	NA	NA	0.853	NA	NA	NA	0.848
0.94	**rx-mle W++**	0.811	NA	NA	NA	0.857	NA	NA	NA	0.878	NA	NA	NA	0.865
0.95	**rx-mle W++**	0.835	NA	NA	NA	0.867	NA	NA	NA	0.89	NA	NA	NA	0.867
0.96	**rx-mle W++**	0.824	NA	NA	NA	0.874	NA	NA	NA	0.891	NA	NA	NA	0.905
0.97	**rx-mle W++**	0.822	NA	NA	NA	0.87	NA	NA	NA	0.895	NA	NA	NA	0.924
0.98	**rx-mle W++**	0.843	NA	NA	NA	0.879	NA	NA	NA	0.929	NA	NA	NA	0.933
0.985	**rx-mle W++**	0.854	NA	NA	NA	0.898	NA	NA	NA	0.934	NA	NA	NA	0.929
0.99	**rx-mle W++**	0.875	NA	NA	NA	0.921	NA	NA	NA	0.939	NA	NA	NA	0.933
0.993	**rx-mle W++**	0.878	NA	NA	NA	0.926	NA	NA	NA	0.96	NA	NA	NA	0.94
0.995	**rx-mle W++**	0.86	NA	NA	NA	0.932	NA	NA	NA	0.955	NA	NA	NA	0.956
0.997	**rx-mle W++**	0.882	NA	NA	NA	0.946	NA	NA	NA	0.957	NA	NA	NA	0.958
0.999	**rx-mle W++**	0.89	NA	NA	NA	0.945	NA	NA	NA	0.958	NA	NA	NA	0.974
0.9999	**rx-mle W++**	0.919	NA	NA	NA	0.974	NA	NA	NA	0.979	NA	NA	NA	0.984
0.99999	**rx-mle W++**	0.915	NA	NA	NA	0.97	NA	NA	NA	0.992	NA	NA	NA	0.993

THIS PAGE INTENTIONALLY LEFT BLANK

APPENDIX C. S-PLUS CODE FOR RIGHT CENSORED AND COMPLETE DATA

The following code is implemented as a function within S-Plus. It pertains to test sets consisting of right censored and complete data. The function computes coverage probabilities for each level of confidence considered, based of Monte Carlo samples of size 1000.

```
function(n, k, beta, eta, alpha) {
# five inputs – sample size, number failures, β, η, (1-α)100%.
        data <- rweibull(n, beta, eta)
# generates n random #'s (time) from the Weibull Dist with parameters β, η.
        data1 <- sort(data)
# sorts data from lowest to highest.
        data.cens <- data1
# duplicates vector data1
        v <- vector(mode = "logical", length = n)
# creates vector "v" denoting whether the time in data1[I] is suspended (0) or failed (1)
        for(i in 1:k) {
                v[i] <- 1
        }
        for(i in (k + 1):n) {
                v[i] <- 0
                data.cens[i] <- data1[k]
                # assigns last (n-k) positions (suspended) to last failure time.
        }

        data2 <- data1[1:k]
# cuts the last (n-k) times (did not fail) for median rank calculations.
        medianrank <- vector(mode = "numeric", length
                = k)
# creates vector medianrank size = k.
        y <- vector(mode = "numeric", length = k)
        x <- vector(mode = "numeric", length = k)
# creates vectors x and y for regression.
        for(i in 1:k) {
                s <- 2 * (n - i + 1)
                t <- 2 * i
                medianrank[i] <- 1/(1 + (((n - i + 1)/i ) * qf(0.5, s, t)))
# compute median ranks (y-axis).
                y[i] <- log( - log(1 - medianrank[i]))
# converts to Weibull linear plot scale (y-axis)
                x[i] <- log(data2[I])   #
# converts to Weibull linear plot scale (x-axis)
```

```
        }
        regression <- lm(x ~ y)
# runs regression x on y (we know y-value and want x-value)
        b1.reg <- predict(regression, data.frame(y = log( - log(1 - 0.01))), se.fit = T)
# predicts B1 based on regression fit and computes standard error.
        s.1 <- survReg(Surv(data.cens, v) ~ 1, dist = "weibull")
# runs MLE on data.cens (length = n) and v (length = n)
        b1.mle <- predict(s.1, data.frame(1), p = c(0.01), type = "uquantile", se = T)
# predicts B1 based on MLE fit and computes standard error.
        tlow.rrx.mle <- b1.reg$fit - qt(alpha, k - 2) * b1.mle$se.fit
# computes RRX-MLE B1LCBα based on regression fit and MLE standard error.
        tlow.rrx.rrx <- b1.reg$fit - qt(alpha, k - 2) * b1.reg$se.fit
# computes RRX-RRX B1LCBα based on regression fit and standard error.
        tlow.mle.mle <- b1.mle$fit - qnorm(alpha) * b1.mle$se.fit
# computes MLE-MLE B1LCBα based on MLE fit and standard error.

        tlow.conf.rrx.mle <- exp(tlow.rrx.mle)
        tlow.conf.rrx.rrx <- exp(tlow.rrx.rrx)
        tlow.conf.mle.mle <- exp(tlow.mle.mle)
# transforms log(time) to time.
        return(c(tlow.conf.rrx.rrx, tlow.conf.rrx.mle, tlow.conf.mle.mle,
                tlow.conf.mle.rrx))
# returns B1LCBα to function for probability coverage calculation.
}
```

APPENDIX D. S-PLUS CODE FOR AAAV APPLICATION

The following code is implemented as a function within S-Plus. It pertains to the specific AAAV test. The function computes MLE-MLE coverage probabilities for each level of confidence considered, based of Monte Carlo samples of size 1000.

```
function(n, k, beta, eta, alpha){
# five inputs – sample size (204), number failures (23), β, η, α.
        data <- rweibull(n, beta, eta)
# generates 204 random #'s (time) from the Weibull Dist with parameters β, η.
        data1 <- sort(data)
# sorts data from lowest to highest.
        for(i in 25:n) {
                data1[i] <- data1[24]
        }
# all suspended items after the 23rd failure had the same suspension times.
        v <- c(1, 1, 1, 1, 1, 1, 1, 1, 0, 0, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1,
                1, 1, 1, 1, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
                0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
                0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
                0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
                0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
                0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
                0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
                0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
                0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
                0, 0, 0, 0)
# creates vector v which specifies where the failures ("1") occur.
        s.1 <- survReg(Surv(data1, v) ~ 1, dist = "weibull")
# runs MLE on vector "v" and vector "data1"
        b1.mle <- predict(s.1, data.frame(1), p = c(0.01), type = "uquantile", se = T)
# predicts B1 based on MLE fit and standard error.
        tlow.mle.mle <- b1.mle$fit - qnorm(alpha) * b1.mle$se.fit
# computes MLE-MLE B1LCBα based on MLE fit and standard error.
        tlow.conf.mle.mle<- exp(tlow.mle.mle)
# transforms log(time) to time
        return(c(tlow.mle.mle))
# returns B1LCBα for computing coverage probability
}
```

THIS PAGE INTENTIONALLY LEFT BLANK

LIST OF REFERENCES

Barlow, R.E., *Engineering Reliability*, American Statistical Association and the Society for Industrial and Applied Mathematics, 1998.

Cook, D., Weisburg, S., *Applied Regression Including Computing and Graphics*, John Wiley and Sons, 1999.

Devore, J.L., *Probability and Statistics for Engineering and the Sciences*, Brooks/Cole Publishing Company, 1995.

Geisser, S., *Predictive Inference: An Introduction*, Chapman and Hall, Inc., 1993.

Ireson, G., Coombs, C., *Handbook of Reliability Engineering and Management*, McGraw-Hill, 1988.

Kales, P., *Reliability for Technology, Engineering, and Management*, Prentice Hall, 1998.

Lawless, J., *Statistical Models and Methods for Lifetime Data*, John Wiley and Sons, 1982.

Meeker, W., Escobar, L., *Statistical Methods for Reliability Data*, John Wiley and Sons, 1998.

Meeker, W., Nelson W., "Tables for the Weibull and Smallest Extreme Value Distributions," General Electric Company Corporate Research and Development TIS Report 74CRD230, 1974.

Menon, M. V., "Estimation of the Shape and Scale Parameters of the Weibull Distribution," *Technometrics* 21, 435-441, 1979.

Nelson, W., *Accelerated Testing – Statistical Models, Test Plans, and Data Analyses,* John Wiley and Sons, 1990.

Nelson, W., *Applied Life Data Analysis*, John Wiley and Sons, 1992.

ReliaSoft Corporation, *Life Data Analysis Reference*, ReliaSoft Publishing, 1997.

Ruble, E., "Validation of Weibull Confidence Bounds," Research Paper, University of Dayton School of Engineering, 1999.

Sorrel, Anthony E., "AIM-7 Sparrow MK-58 Rocket Motor Reliability and Life Data Analysis." Master's Thesis, Operations Research Department, Naval Post Graduate School, 2000.

THIS PAGE INTENTIONALLY LEFT BLANK

INITIAL DISTRIBUTION LIST

No. of Copies

1. Defense Technical Information Center 2
   8725 John. J. Kingman Road STE 0944
   Fort Belvoir, VA 22060-6218

2. Dudley Knox Library.. 2
   Naval Postgraduate School
   411 Dyer Road
   Monterey, CA 93943-5101

3. Director, Training and Eduacation 1
   MCCDC, Code C46
   1019 Elliot Road
   Quantico, VA 22134-5027

4. Director, Marine Corps Research Center 3
   MCCDC, Code C40RC
   101 Strauss Avenue
   Indian Head, MD 20640-5000

5. Marine Corps Representative 1
   Naval Post Graduate School
   Code 037, Bldg. 330, Ingersoll Hall, Room 116
   555 Dyer Road
   Monterey, CA 93943

6. Marine Corps Tactical Systems Support Activity 1
   Technical Advisory Branch
   Attn: Librarian
   Box 555171
   Camp Pendleton, CA 92055-5080

7. Director, Studies and Analysis Division 2
   MCCDC Code C45
   300 Russell Road
   Quantico, VA 22134-5130

8. Captain Jon L. Halverson 3
   HQMC, DC/S I&L (LS)
   2 Navy Annex
   Washington D.C. 20380-1775,

9. LTC Dave H. Olwell, Code OR/Ol 4
Department of Operation Research
Naval Postgraduate School
Monterey, CA 93943-5219

10. Robert R. Read OR/Re 2
Department of Operation Research
Naval Postgraduate School
Monterey, CA 93943-5219